

Lernen Motorischer Fähigkeiten: Von Algorithmen zu Roboter-Experimenten *

Jens Kober

CoR-Lab - Universität Bielefeld
Honda Research Institute Europe
jkober@cor-lab.uni-bielefeld.de

Abstract: In dieser Doktorarbeit werden Ansätze diskutiert, die es Robotern ermöglichen, motorische Fähigkeiten zu erlernen. Motorische Fähigkeiten können oft durch Motor-Primitive, die elementare Bewegungen kodieren, dargestellt werden. Es gab schon eine Reihe von erfolgreichen Anwendungen des Erlernens von Motor-Primitiven durch überwachtetes Lernen. Allerdings sind viele interessante motorische Lernprobleme hochdimensionale Reinforcement-Learning-Probleme (bestärkendes Lernen), die oft außerhalb der Anwendbarkeit der aktuellen Reinforcement-Learning-Methoden liegen. Diese Doktorarbeit leistet einen Beitrag zum Stand der Technik in Reinforcement-Learning, angewandt auf die Robotik, sowohl durch neuartige Algorithmen als auch durch neuartige Anwendungen. Gezeigt wird, wie motorische Fähigkeiten auf drei verschiedenen Ebenen gelernt werden können. Alle vorgeschlagenen Ansätze wurden ausgiebig mit Aufgaben wie u. a. einem Becherspiel, Darts, Tischtennis oder einem Wurfspiel in der Simulation und auf realen Robotern validiert.

1 Einführung

Seit Issac Asimov in den 1940er Jahren begann Kurzgeschichten über Roboter zu schreiben, werden Roboter von der Öffentlichkeit als potentielle Haushaltshilfen, Begleiter und Soldaten angesehen. Science-Fiction-Filme zeigen Roboter sowohl als Freunde als auch als Feinde der Menschheit, aber in beiden Fällen liegen deren Fähigkeiten weit über den Fähigkeiten der aktuellen realen Roboter. Simon [Sim65], einer der Pioniere der künstlichen Intelligenz (KI), behauptete, dass „Maschinen innerhalb von zwanzig Jahren in der Lage sein werden, jegliche Arbeit, die ein Mensch verrichten kann, zu erledigen.“ Solche über-optimistischen Versprechen führten dazu, dass heutige Prognosen eher konservativ ausfallen. Im Moment werden Roboter langsam Teil unseres täglichen Lebens in Form von Spielzeugen und Haushaltshilfen, wie autonome Staubsauger, Rasenmäher und Fensterputzer. Die meisten anderen Roboter sind noch auf Forschungslabore und industrielle Umgebungen beschränkt. Viele Aufgaben des täglichen Lebens können nur sehr langsam von einem Roboter verrichtet werden und die Fähigkeit zur Verallgemeinerung ist oft sehr begrenzt. Daher sind alle diese Systeme noch weit von den Erwartungen, die von Literatur

* Englischer Titel der Dissertation: "Learning Motor Skills: From Algorithms to Robot Experiments" [Kob12]



Abbildung 1: Diese Abbildung illustriert die Verallgemeinerung eines Rückhand-Motor-Primitives. Der Roboter verwendet jeweils dasselbe Motor-Primitiv, um auf Bälle an unterschiedlichen Positionen mit unterschiedlichen Geschwindigkeiten zu reagieren. Hierfür werden nur die Meta-Parameter des Motor-Primitives angepasst.

und Filmen geweckt werden, als auch von den Träumen der KI-Forscher entfernt.

Insbesondere in Japan wurde die Notwendigkeit von Haushalts-Robotern aufgrund der Alterung der Bevölkerung erkannt. Eine der wichtigsten Herausforderungen ist, mit sich ändernden Situationen in einer von Menschen und Robotern geteilten Umgebung umzugehen (z. B. umgestellte Möbel, wechselnde Lichtverhältnisse, kooperative Aufgaben) und die Notwendigkeit, sich den individuellen Anforderungen und Erwartungen der menschlichen Besitzer anzupassen. Die meisten aktuell käuflich erwerblichen Produkte verfolgen entweder einen Universal-Ansatz, der oft nicht optimal ist (z. B. Staubsauger-Roboter, die ihre Umgebung nicht wahrnehmen, sondern einen Algorithmus verwenden, der auf Hindernisse reagiert und der eine Abdeckung des gesamten Bodens garantiert [Bot12]) oder einen Ansatz, der einen Setup-Schritt entweder in Software (z. B. Eingabe eines Raumplans) oder in Hardware (z. B. Anbringen von Markern) erfordert. Als Alternative könnte man sich ein selbst-lernendes System vorstellen.

Diese Doktorarbeit behandelt keine Navigations-Probleme, sondern konzentriert sich auf das Erlernen von motorischen Fähigkeiten [Wul07]. In erster Linie werden Aufgaben betrachtet, bei denen das dynamische Verhalten des Roboters und seiner Umgebung berücksichtigt werden muss und wo ein kinematischer Bewegungsplan nicht ausreichend ist. Motorische Fähigkeiten können oft durch Motor-Primitive dargestellt werden. Solche Motor-Primitive kodieren elementare Bewegungen, die verallgemeinert, aneinandergereiht und kombiniert werden können, um komplexere Aufgaben zu erfüllen. Zum Beispiel könnte man ein Vorhand- und Rückhand-Spiel als zwei verschiedene Motor-Primitive für Tischtennis ansehen, welche je nach Ballposition verallgemeinert werden (siehe Abbildung 1). Die Arbeit fokussiert auf das Lernen der optimalen Ausführung von Motor-Primitiven und wie man solche auf neue Situationen verallgemeinern kann. Die vorgestellten Anwendungen kommen aus dem Sport- und Spiel-Bereich, aber die vorgestellten Techniken können auch bei alltäglichen Aufgaben im Haushalt Anwendung finden.

2 Motivation

Die Forschung an Ansätzen des maschinellen Lernens hat zu einer Vielzahl von Algorithmen geführt. Leider sind die meisten Standard-Ansätze nicht direkt auf die Robotik anwendbar, vor allem aufgrund der inhärenten hohen Dimensionalität. Daher besteht ein Bedarf an

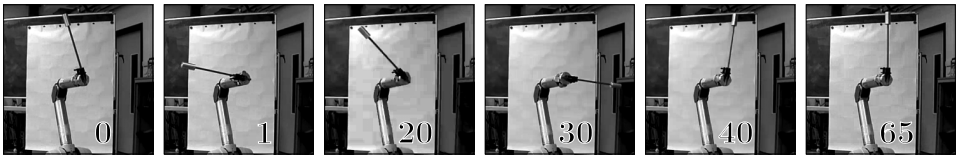


Abbildung 2: Diese Abbildung illustriert den Lernprozess des „Underactuated Swing-Up“. Ziel dieser Aufgabe ist es, das schwere Pendel von einer hängenden Position nach oben zu schwingen und dort zu balancieren. Das Drehmoment der Motoren ist limitiert, damit das Pendel nicht direkt nach oben bewegt werden kann, sondern vorher Schwung geholt werden muss. Die Bilder zeigen jeweils die Position am Ende einer Iteration. Der Lernprozess wurde mit einer Policy (Strategie), die durch Imitations-Lernen erlangt wurde, initialisiert (Bild 0). Bild 1 zeigt dieselbe Policy mit aktivierter Drehmoment-Limitierung.

Methoden, die spezifisch auf die Bedürfnisse des Roboter-Lernens zugeschnitten sind. Diese Doktorarbeit fokussiert auf das Reinforcement-Learning (bestärkende Lernen) für Motor-Primitive.

Die Reinforcement-Learning-Forschung begann in den 1950er Jahren (z. B. [Min54, FC54, Bel57]), betrachtete aber vor allem theoretische Probleme. Algorithmen wurden häufig auf synthetischen Benchmark-Problemen mit diskreten Zuständen und Aktionen evaluiert. Die wohl bekanntesten realen Anwendungen des Reinforcement-Learning sind Spiele wie Backgammon [Tes94] oder Go [CKL96], aber auch Roboter-Anwendungen können schon in den 1990er Jahren gefunden werden (z. B. [MC92, GFB94]). Im Gegensatz zu vielen Problemen in der Reinforcement-Learning-Literatur haben Roboter-Probleme inhärent kontinuierliche Zustände und Aktionen. Darüber hinaus benötigen Experimente in der Robotik oft teure und potenziell empfindliche Hardware und erfordern auch oft menschliche Überwachung und menschliches Eingreifen. Diese Unterschiede erfordern die Anpassung bestehender Reinforcement-Learning-Ansätze oder die Entwicklung von maßgeschneiderten Ansätzen.

Policy-Search (Strategie-Suche) hat sich als eine Alternative zum wertefunktion-basierten Reinforcement-Learning [SM01, KHS01, Pes01, BKNS04, EFCR06, TWS07] etabliert. Speziell für das Lernen von motorischen Fähigkeiten in der Robotik ist die direkte Suche nach einer Policy vielversprechend, statt zunächst das duale Problem (d. h. die Bestimmung der Werte-Funktion) zu lösen. Darüber hinaus können die Integration von Vorwissen in Form der Policy-Struktur, einer anfänglichen Policy oder eines Modells des Systems die benötigte Lernzeit drastisch reduzieren. Diese Doktorarbeit behandelt Policy-Search-Methoden, die eine vorstrukturierte Policy und eine anfängliche Policy nutzen können.

3 Beiträge

Diese Doktorarbeit leistet einen Beitrag zum Stand der Technik in Reinforcement-Learning, angewandt auf die Robotik, sowohl durch neuartige Algorithmen (Abschnitt 3.1) als auch durch neuartige Anwendungen (Abschnitt 3.2).

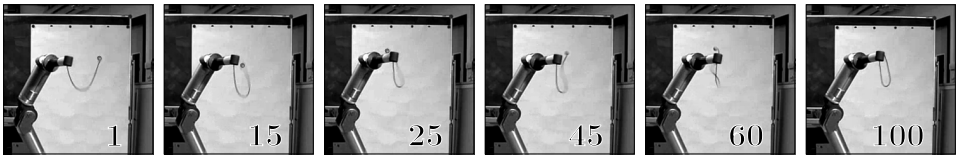


Abbildung 3: Diese Abbildung illustriert den Lernprozess der Becherspiel-Aufgabe. Das Ziel dieser Aufgabe ist es, den Ball in dem Becher zu fangen. Die Bilder zeigen jeweils die Position des Balls, die dem Becher pro Iteration am nächsten kam. Der Lernprozess wurde mit einer Policy, die durch Imitationslernen erlangt wurde, initialisiert. Dieses Nachmachen alleine genügt nicht, um die Aufgabe erfolgreich zu lösen (ganz links). Nach etwa 100 Versuchen hat der Roboter gelernt, den Ball zuverlässig zu fangen (ganz rechts).

3.1 Algorithmen

Reinforcement-Learning, angewandt auf die Robotik, muss mit einer Reihe von Herausforderungen umgehen, wie im zweiten Kapitel der Arbeit anhand einer Literatur-Übersicht diskutiert wird. Diese Studie liefert einen Überblick über die verschiedenen Techniken und möglichen Anwendungen. Der Fokus liegt auf den spezifischen Herausforderungen und Ansätzen, die das Lern-Problem trotzdem lösbar machen. Die Dimensionalität ist systembedingt hoch und kontinuierlich. Iterationen in der realen Welt sind teuer, da sie Zeit und menschliche Betreuung benötigen und potenziell teure und empfindliche Hardware voraussetzen. Der Einsatz von Modellen kann diese Probleme reduzieren, erzeugt jedoch andere Herausforderungen. Jeder Lernalgorithmus ist durch die Qualität der Kosten-Funktion beschränkt. Das Aufstellen einer Kosten- oder einer Belohnungs-Funktion ist oft nicht einfach. Selbst die Kostenfunktion einer einfachen menschlichen Greifbewegung ist nicht vollständig geklärt [BW07]. Inverses Reinforcement-Learning ist eine vielversprechende Alternative zur manuellen Spezifizierung der Belohnungs-Funktion und wurde für die Becherspiel-Aufgabe untersucht [BKP11].

Idealerweise sollte ein Algorithmus, der auf einen Roboter angewandt wird, sicher sein, d. h. eine Beschädigung des Roboters muss vermieden werden und der Algorithmus sollte schnell sein, sowohl im Hinblick auf die Konvergenz als auch auf die Rechenzeit. Wichtige Punkte für schnelle Konvergenz eines Algorithmus sind, dass er Versuche effizient nutzt, nur sehr wenige offene Parameter hat und Vorwissen nutzen kann. In dieser Doktorarbeit werden innerhalb eines Frameworks der erfolgsgewichteten Nachahmung Algorithmen vorgestellt, die diesen Anforderungen entsprechen.

Das Übersichts-Kapitel „Reinforcement Learning in Robotics: A Survey“ basiert auf [KBP13] und einer vorhergehenden Version [KP12b].

3.1.1 Policy learning by Weighting Exploration with the Returns (PoWER)

Im Kapitel „Policy Search for Motor Primitives in Robotics“ der Arbeit wird der „Policy learning by Weighting Exploration with the Returns“-Algorithmus (PoWER) vorgestellt. PoWER ist eine erwartungswert-maximierungs-inspirierte Policy-Search-



Abbildung 4: Diese Abbildung illustriert einen Dart-Wurf mit dem Humanoiden JST-ICORP/SARCOS CBi. Der Roboter hat die Aufgabe, einen gelernten, erfolgreichen Wurf so zu verallgemeinern, dass alle Felder auf der Dartscheibe zuverlässig getroffen werden können.

Methode, die auf der strukturierten Exploration im Parameter-Raum beruht. In diesem Kapitel wird ein Framework erfolgsgewichteter Nachahmung eingeführt. Basierend auf [DH97], wird der Return (Gewinn) einer Episode als eine nicht-normalisierte Wahrscheinlichkeitsverteilung betrachtet. Somit kann eine untere Schranke des Logarithmus des erwarteten Returns gefunden und maximiert werden. Je nach Optimierungs-Strategie dieser unteren Schranke und der Explorations-Strategie entspricht das Framework mehreren bekannten Policy-Search-Methoden: dem episodischen REINFORCE [Wil92], dem „Policy Gradient Theorem“ [SMSM99], dem episodischen „Natural Actor Critic“ [PS08b], sowie einer Verallgemeinerung der „Reward-Weighted Regression“ [PS08a]. Der neuartige Algorithmus, PoWER, basiert auf einer erwartungswert-maximierungs-inspirierten Optimierung und einer strukturierten, zustands-abhängigen Explorations-Strategie. Dieser Ansatz hat bereits in anderen Publikationen Anwendungen gefunden, z. B. [KCC10, KUC⁺11, NFV⁺12, dSKB12, VOG⁺13]. Algorithmen, die quasi identisch zu PoWER sind, können auch über Ansätze wie „Path Integrals“ [TBS10], „Relative Entropy Policy Search“ [PMA10] und „Monte-Carlo EM“-Algorithmen [VTKP09] hergeleitet werden.

Die zugehörigen Experimente sind im Abschnitt 3.2.2 beschrieben. Dieses Kapitel basiert auf [KP11b] und den vorläufigen Versionen [KMP08, KP08, KP09, KP10].

3.1.2 Cost-regularized Kernel Regression (CrKR)

Im Kapitel „Reinforcement Learning to Adjust Parametrized Motor Primitives to New Situations“ der Arbeit wird der Algorithmus „Cost-regularized Kernel Regression“ (CrKR) vorgestellt. CrKR ist eine nicht-parametrische Policy-Search-Methode, die sich besonders für das Lernen von Meta-Parametern, d. h. einer kleinen Anzahl von Parametern, die die Bewegung global beeinflussen, eignet. In diesem Szenario kann es schwierig sein, eine gute Parametrisierung für die Policy zu finden; hier bietet eine nicht-parametrische Policy mehr Flexibilität. CrKR wird basierend auf der „Reward-Weighted Regression“ [PS08a] hergeleitet. Der resultierende Algorithmus ist mit der Gaussian-Process-Regression (Gauß-Prozess-Regression) verwandt und ähnlich zu dieser hat er auch eine prädiktive Unsicherheit, die dazu eingesetzt wird, die Exploration des Reinforcement-Learnings zu leiten. Dieser Ansatz wird verwendet, um eine Funktion zu lernen, die die Meta-Parameter abhängig von der momentanen Situation ergibt. Die Parameter der Motor-Primitive, die die Feinheiten der Bewegung bestimmen, können vorher durch andere Verfahren gelernt werden.

Die zugehörigen Experimente sind im Abschnitt 3.2.3 beschrieben. Dieses Kapitel basiert auf [KWOP12] und den vorläufigen Versionen [KOP10, KOP11, KP11a].

3.2 Anwendungen

Diese Doktorarbeit zeigt eine große Anzahl von Benchmark-Aufgaben und untersucht die vorgestellten Ansätze mit Roboter-Aufgaben, sowohl mit simulierten als auch mit realen Robotern. Die eingesetzten Roboter sind ein Barrett WAM, ein BioRob, der Humanoide JST-ICORP/SARCOS CBi und ein Kuka KR 6. Diese Dissertation untersucht, wie einzelne motorische Fähigkeiten gelernt werden können und wie man diese verallgemeinern kann, um sie in neuen Situationen anzuwenden. Die vorliegende Arbeit hat zum Stand des Reinforcement-Learnings, angewandt auf die Robotik, durch die Erforschung hochdynamischer Bewegungen und die Vorstellung sehr effizienter Lernprozesse beigetragen.



Abbildung 5: Diese Abbildung illustriert das Setup für das Wurf-Spiel. Der Roboter lernt gleichzeitig, die Ziele zu treffen und ein Spiel zu spielen, das von Black Jack inspiriert wurde.

3.2.1 Darstellung Motorischer Fähigkeiten

Das Kapitel „Movement Templates for Learning of Hitting and Batting“ der Doktorarbeit beschreibt, wie die „Dynamical Systems Motor Primitives“-Darstellung [INS02] für motorische Fähigkeiten verallgemeinert werden kann, um Schlag-Bewegungen, wie sie z. B. bei Tischtennis benötigt werden (Abbildung 1), zu repräsentieren. Dieses Kapitel basiert auf [KMK⁺10].

3.2.2 Einzelne Motorische Fähigkeiten

Aufgrund der hohen Dimensionalität können motorische Fähigkeiten nicht von Null an gelernt werden. Stattdessen wird imitiert wie Kinder lernen, und der Lernprozess durch Imitations-Lernen initialisiert. Beispielsweise kann der Roboter durch die Bewegung geführt werden, um eine anfängliche Policy zu erhalten. Anschließend lernt der Reinforcement-Learning-Algorithmus, diese Fähigkeit zuverlässiger auszuführen. Nach einer realistischen Anzahl von Iterationen (siehe Abbildungen 2 und 3) kann die Aufgabe regelmäßig erfüllt werden und der Roboter zeigt eine hervorragende durchschnittliche Leistung. Dieser Ansatz wird mit einer Reihe von verschiedenen Policy-Parametrisierungen, u. a. mit den „Dynamical Systems Motor Primitives“ [INS02] und anderen parametrischen Darstellungen, untersucht. PoWER wird mit einer Vielzahl von Policy-Search-Ansätzen verglichen, sowohl mit einfachen Beispielen als auch mit Roboter-Aufgaben. Der „Underactuated Swing-Up“ (Abbildung 2) und eine komplexe Bechenspiel-Aufgabe (Abbildung 3) werden auf einem echten Barrett WAM effizient gelernt.



Abbildung 6: Diese Abbildung illustriert die gelernte Tischtennisball-Jonglier-Fähigkeit. Die Aufgabe ist es, den Ball auf dem Schläger hüpfen zu lassen. Hierzu musste der Lernalgorithmus bestimmen, welche Motor-Primitive Vorrang haben, z. B. ob es wichtiger ist, den Ball genau mit der Mitte des Schlägers zu treffen oder mit der richtigen Orientierung.

3.2.3 Verallgemeinerte Motorische Fähigkeiten

Um die Effizienz eines Lernprozesses zu erhöhen, ist es häufig vorteilhaft, eine motorische Fähigkeit an neue Situationen anzupassen, statt sie komplett neu zu lernen. Diese Art des Lernens erfordert oft eine nicht-intuitive Zuordnung von Situation zu Aktionen. Um den vorgeschlagenen Ansatz in einem komplexen Szenario zu demonstrieren, wurden das „Around the Clock“-Dart-Spiel (Abbildung 4), Tischtennis (Abbildung 1) und ein Wurf-Spiel (Abbildung 5) gewählt und diese sowohl auf simulierten als auch auf realen Robotern implementiert. In diesen Szenarien wird gezeigt, dass der Ansatz mit einer Vielzahl von Situationen zurecht kommt, d. h. unterschiedlichen Kosten-Funktionen (mit und ohne sekundären Zielen) und ganz unterschiedlichen Policies mit ihren zugehörigen Meta-Parametern. Die Experimente wurden auf vier verschiedenen realen Robotern durchgeführt (einem Barrett WAM, einem BioRob, dem Humanoiden JST-ICORP/SARCOS CBi und einem Kuka KR 6). Das Wurf-Spiel zeigt erste Schritte in Richtung eines hierarchischen Reinforcement-Learning-Systems.

Viele Roboter-Aufgaben können in Teilaufgaben zerlegt werden. Doch oft sind diese Teilaufgaben nicht gleichzeitig erfüllbar und die Prioritäts-Struktur muss bestimmt werden. Erste Schritte in Richtung eines Regler-Gesetzes, das auf einer Anzahl von priorisierten Motor-Primitiven beruht, werden mit einer Tischtennisball-Jonglier-Aufgabe (siehe Abbildung 6) auf einem Barrett WAM untersucht. Das Kapitel „Learning Prioritized Control of Motor Primitives“ basiert auf [KP12a].

4 Fazit

Diese Doktorarbeit erweitert den Stand der Forschung in Reinforcement-Learning, angewandt auf die Robotik, sowohl durch neuartige Algorithmen als auch durch neuartige Anwendungen. Die vorgestellten Algorithmen, die aus dem neuen Framework erfolgsgewichteter Nachahmung entstammen, entsprechen den speziellen Anforderungen des Lernens auf realen Robotern, die in Abschnitt 3.1 erwähnt wurden. Alle vorgeschlagenen Ansätze wurden ausführlich mit bestehenden Ansätzen verglichen und mit herausfordernden Roboter-Aufgaben validiert. Diese sind unter anderem die komplexe Becherspiel-Aufgabe,

das Verallgemeinern von Dart-Würfen und Tischtennis-Schlägen, ein Zielwurf-Spiel und eine Tischtennisball-Jonglier-Aufgabe. Die Ansätze wurden mit verschiedenen realen Robotern realisiert: mit einem Barrett WAM, dem Humanoiden JST-ICORP/SARCOS CBi, einem KUKA KR 6 Roboter und einem BioRob.

Literatur

- [Bel57] R. E. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, NJ, 1957.
- [BKNS04] J. A. Bagnell, S. Kadade, A. Ng und J. C. Schneider. Policy Search by Dynamic Programming. In *Advances in Neural Information Processing Systems (NIPS)*, 2004.
- [BKP11] A. Boularias, J. Kober und J. Peters. Relative Entropy Inverse Reinforcement Learning. *Journal of Machine Learning Research - Proceedings Track*, 15:182–189, 2011.
- [Bot12] BotJunkie. BotJunkie Interview: Nancy Dussault Smith On iRobot’s Roomba, Februar 2012.
- [BW07] P. M. Bays und D. M. Wolpert. Computational Principles of Sensorimotor Control that Minimise Uncertainty and Variability. *Journal of Physiology*, 578:387–396, 2007.
- [CKL96] H. W. Chan, I. King und J. C. S. Lui. Performance Analysis of a New Updating Rule for TD(λ) Learning in Feedforward Networks for Position Evaluation in Go Game. In *IEEE Int. Conf. on Neural Networks*, 1996.
- [DH97] P. Dayan und G. E. Hinton. Using Expectation-Maximization for Reinforcement Learning. *Neural Computation*, 9(2):271–278, 1997.
- [dSKB12] B. C. da Silva, G. D. Konidaris und A. G. Barto. Learning Parameterized Skills. In *Int. Conf. on Machine Learning*, 2012.
- [EFCR06] A. El-Fakdi, M. Carreras und P. Ridao. Towards Direct Policy Search Reinforcement Learning for Robot Control. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2006.
- [FC54] B. G. Farley und W. A. Clark. Simulation of Self-organizing Systems by Digital Computer. *IRE Transactions on Information Theory*, 4:76–84, 1954.
- [GFB94] V. Gullapalli, J. Franklin und H. Benbrahim. Acquiring Robot Skills via Reinforcement Learning. *IEEE Control Systems Journal, Special Issue on Robotics: Capturing Natural Motion*, 4(1):13–24, 1994.
- [INS02] A. J. Ijspeert, J. Nakanishi und S. Schaal. Learning Attractor Landscapes for Learning Motor Primitives. In *Advances in Neural Information Processing Systems (NIPS)*, 2002.
- [KBP13] J. Kober, J. A. Bagnell und J. Peters. Reinforcement Learning in Robotics: A Survey. *Int. Journal of Robotics Research*, 2013.
- [KCC10] P. Kormushev, S. Calinon und D. G. Caldwell. Robot Motor Skill Coordination with EM-based Reinforcement Learning. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2010.
- [KHS01] I. Kwee, M. Hutter und J. Schmidhuber. Gradient-based Reinforcement Planning in Policy-Search Methods. In *Europ. Workshop on Reinforcement Learning (EWRL)*, 2001.

- [KMK⁺10] J. Kober, K. Mülling, O. Krömer, C. H. Lampert, B. Schölkopf und J. Peters. Movement Templates for Learning of Hitting and Batting. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2010.
- [KMP08] J. Kober, B. Mohler und J. Peters. Learning Perceptual Coupling for Motor Primitives. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2008.
- [Kob12] J. Kober. *Learning Motor Skills: From Algorithms to Robot Experiments*. Dissertation, TU Darmstadt, April 2012.
- [KOP10] J. Kober, E. Oztop und J. Peters. Reinforcement Learning to Adjust Robot Movements to New Situations. In *Robotics: Science and Systems Conf. (R:SS)*, 2010.
- [KOP11] J. Kober, E. Oztop und J. Peters. Reinforcement Learning to Adjust Robot Movements to New Situations. In *Int. Joint Conf. on Artificial Intelligence (IJCAI), Best Paper Track*, 2011.
- [KP08] J. Kober und J. Peters. Policy Search for Motor Primitives in Robotics. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.
- [KP09] J. Kober und J. Peters. Learning Motor Primitives for Robotics. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2009.
- [KP10] J. Kober und J. Peters. Imitation and Reinforcement Learning - Practical Algorithms for Motor Primitive Learning in Robotics. *IEEE Robotics and Automation Magazine*, 17(2):55–62, 2010.
- [KP11a] J. Kober und J. Peters. Learning Elementary Movements Jointly with a Higher Level Task. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2011.
- [KP11b] J. Kober und J. Peters. Policy Search for Motor Primitives in Robotics. *Machine Learning*, 84(1-2):171–203, 2011.
- [KP12a] J. Kober und J. Peters. Learning Prioritized Control of Motor Primitives. 2012. [arXiv:1209.0488](https://arxiv.org/abs/1209.0488) [cs.LG].
- [KP12b] J. Kober und J. Peters. *Reinforcement Learning: State-of-the-Art*, Kapitel Reinforcement Learning in Robotics: A Survey, Seiten 579–610. Springer Berlin / Heidelberg, 2012.
- [KUC⁺11] P. Kormushev, B. Ugurlu, S. Calinon, N. Tsagarakis und D. G. Caldwell. Bipedal Walking Energy Minimization by Reinforcement Learning with Evolving Policy Parameterization. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2011.
- [KWOP12] J. Kober, A. Wilhelm, E. Oztop und J. Peters. Reinforcement Learning to Adjust Parametrized Motor Primitives to New Situations. *Autonomous Robots*, 33(4):361–379, 2012.
- [MC92] S. Mahadevan und J. Connell. Automatic Programming of Behavior-based Robots using Reinforcement Learning. *Artificial Intelligence*, 55(2-3):311–365, 1992.
- [Min54] M. L. Minsky. *Theory of Neural-Analog Reinforcement Systems and its Application to the Brain-Model Problem*. Dissertation, Princeton University, 1954.
- [NFV⁺12] B. Nemeč, D. Forte, R. Vuga, M. Tamosiunaite, F. Wörgötter und A. Ude. Applying Statistical Generalization to Determine Search Direction for Reinforcement Learning of Movement Primitives. In *IEEE/RAS Int. Conf. on Humanoid Robots (Humanoids)*, 2012.
- [Pes01] L. Peshkin. *Reinforcement Learning by Policy Search*. Dissertation, Brown University, Providence, RI, 2001.

- [PMA10] J. Peters, K. Mülling und Y. Altun. Relative Entropy Policy Search. In *National Conf. on Artificial Intelligence (AAAI)*, 2010.
- [PS08a] J. Peters und S. Schaal. Learning to Control in Operational Space. *Int. Journal of Robotics Research*, 27(2):197–212, 2008.
- [PS08b] J. Peters und S. Schaal. Natural Actor-Critic. *Neurocomputing*, 71(7-9):1180–1190, 2008.
- [Sim65] H. A. Simon. *The Shape of Automation for Men and Management*. New York: Harper & Row, 1965.
- [SM01] M. Strens und A. Moore. Direct Policy Search using Paired Statistical Tests. In *Int. Conf. on Machine Learning (ICML)*, 2001.
- [SMSM99] R. S. Sutton, D. McAllester, S. Singh und Y. Mansour. Policy Gradient Methods for Reinforcement Learning with Function Approximation. In *Advances in Neural Information Processing Systems (NIPS)*, 1999.
- [TBS10] E. A. Theodorou, J. Buchli und S. Schaal. Reinforcement Learning of Motor Skills in High Dimensions: A Path Integral Approach. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2010.
- [Tes94] G. Tesauro. TD-Gammon, a Self-Teaching Backgammon Program, Achieves Master-Level Play. *Neural Computation*, 6(2):215–219, 1994.
- [TWS07] M. E. Taylor, S. Whiteson und P. Stone. Transfer via Inter-Task Mappings in Policy Search Reinforcement Learning. In *Int. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*, 2007.
- [VOG⁺13] R. Vuga, M. Ogrinc, A. Gams, T. Petrič, N. Sugimoto, A. Ude und J. Morimoto. Motion Capture and Reinforcement Learning of Dynamically Stable Humanoid Movement Primitives. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2013.
- [VTKP09] N. Vlassis, M. Toussaint, G. Kontes und S. Piperidis. Learning Model-free Robot Control by a Monte Carlo EM Algorithm. *Autonomous Robots*, 27(2):123–130, 2009.
- [Wi92] R. J. Williams. Simple Statistical Gradient-following Algorithms for Connectionist Reinforcement Learning. *Machine Learning*, 8:229–256, 1992.
- [Wu07] G. Wulf. *Attention and motor skill learning*. Human Kinetics, Champaign, IL, 2007.



Jens Kober wurde am 6. Mai 1982 in Künzelsau geboren. In 2008 absolvierte er sein Ingenieurstudium an der Universität Stuttgart (Diplom-Ingenieur Technische Kybernetik) und zeitgleich ein französisches Ingenieurstudium (Ecole Centrale Paris). Die Doktorarbeit von Jens Kober wurde im gemeinsamen Roboter-Lern-Labor des Fachgebietes Intelligente Autonome Systeme, TU Darmstadt, und des Max Planck Instituts für Intelligente Systeme, Tübingen, angefertigt. Betreut wurde die Promotion von Professor Dr. Jan Peters. Während seiner Doktorarbeit nutzte Jens Kober die Möglichkeit als Gastwissenschaftler am Advanced Telecommunication Research (ATR) Center in Japan und als Praktikant bei Disney Research Pittsburgh, USA, zu forschen. Seit kurzem

arbeitet er am Offenbacher Honda Research Institute Europe und zugleich am CoR-Lab der Universität Bielefeld.